

What is claimed is:

1 1. An apparatus comprising:
2 a first processor to execute a main thread instruction stream that includes a delinquent
3 instruction;
4 a second processor to execute a helper thread instruction stream that includes a subset of
5 the main thread instruction stream, wherein the subset includes the delinquent instruction;
6 wherein said first and second processors each include a private data cache;
7 a shared memory system coupled to said first processor and to said second processor; and
8 logic to retrieve, responsive to a miss of requested data for the delinquent instruction in
9 the private cache of the second processor, the requested data from the shared memory
10 system;
11 the logic further to provide the requested data to the private data cache of the first
12 processor.

1 2. The apparatus of claim 1, wherein:
2 the first processor, second processor and logic are included within a chip package.

1 3. The apparatus of claim 1, wherein:
2 the shared memory system includes a shared cache.

1 4. The apparatus of claim 3, wherein:
2 the shared memory system includes a second shared cache.

1 5. The apparatus of claim 3, wherein:
2 the shared cache is included within a chip package.

1 6. The apparatus of claim 1, wherein:
2 the logic is further to provide the requested data from the shared memory system to the
3 private data cache of the second processor.

1 7. The apparatus of claim 1, wherein:
2 said first and second processors are included in a plurality of n processors, where $n > 2$;
3 each of said plurality of processors is coupled to the shared memory system; and
4 each of said n plurality of processors includes a private data cache.

1 8. The apparatus of claim 7, wherein:
2 the logic is further to provide the requested data from the shared memory system to each
3 of the n private data caches.

1 9. The apparatus of claim 7, wherein:

2 the logic is further to provide the requested data from the shared memory system to a
3 subset of the n private data caches, the subset including x of the n private data caches, where
4 $0 < x < n$.

1 10. The apparatus of claim 1, wherein:

2 the first processor is further to trigger the second processor's execution of the helper
3 thread instruction stream responsive to a trigger instruction in the main thread instruction
4 stream.

1 11. An apparatus comprising:

2 a first processor to execute a main thread instruction stream that includes a delinquent
3 instruction;

4 a second processor to execute a helper thread instruction stream that includes a subset of
5 the main thread instruction stream, wherein the subset includes the delinquent instruction;

6 wherein said first and second processors each include a private data cache; and

7 logic to retrieve, responsive to a miss of requested data for the delinquent instruction in a
8 first one of the private data caches, the requested data from the other private data cache if
9 said requested data is available in the other private data cache;

10 the logic further to provide the requested data to the first private data cache.

1 12. The apparatus of claim 11, further comprising:

2 a shared memory system coupled to said first processor and to said second processor;
3 wherein said logic is further to retrieve the requested data from the shared memory
4 system if the requested data is not available in the other private data cache.

1 13. The apparatus of claim 11, wherein:

2 the logic is included within an interconnect, wherein the interconnect is to provide
3 networking logic for communication among the first processor, the second processor, and the
4 shared memory system.

1 14. The apparatus of claim 13, wherein:

2 the first and second processor are each included in a plurality of n processors; and
3 the interconnect is further to concurrently broadcast a request for the requested data to
4 each of the n processors and to the shared memory system.

1 15. The apparatus of claim 11, wherein:

2 the memory system includes a shared cache.

1 16. The apparatus of claim 15, wherein:

2 the memory system includes a second shared cache.

1 17. The apparatus of claim 11, wherein:

2 the first processor is further to trigger the second processor's execution of the helper
3 thread instruction stream responsive to a trigger instruction in the main thread instruction
4 stream

1 18. A method comprising:

2 determining that a helper core has suffered a miss in a private cache for a load instruction
3 while executing a helper thread; and
4 prefetching load data for the load instruction into a private cache of a main core.

1 19. The method of claim 18, wherein prefetching further comprises:

2 retrieving the load data from a shared memory system; and
3 providing the load data to the private cache of the main core.

1 20. The method of claim 18, further comprising:

2 providing load data for the load instruction from a shared memory system into the private
3 cache of the helper core.

1 21. The method of claim 18, further comprising:

2 providing load data for the load instruction from a shared memory system into the private
3 cache for each of a plurality of helper cores.

1 22. The method of claim 18, wherein prefetching further comprises:

2 retrieving the load data from a private cache of a helper core; and

3 providing the load data to the private cache of the main core.

1 23. The method of claim 18, wherein prefetching further comprises:

2 concurrently:

3 broadcasting a request for the load data to each of a plurality of cores; and

4 requesting the load data from a shared memory system.

1 24. The method of claim 23, wherein prefetching further comprises:

2 providing, if the load data is available in a private cache of one of the plurality of
3 cores, the load data to the main core from the private cache of one of the plurality of
4 cores ; and

5 providing, if the load data is not available in a private cache of one of the plurality of
6 cores, the load data to the main core from the shared memory system.

1 25. An article comprising:

2 a machine-readable storage medium having a plurality of machine accessible instructions,
3 which if executed by a machine, cause the machine to perform operations comprising:

4 determining that a helper core has suffered a miss in a private cache for a load
5 instruction while executing a helper thread; and

6 prefetching load data for the load instruction into a private cache of a main core.

1 26. The article of claim 25, wherein:

2 the instructions that cause the machine to prefetch load data further comprise instructions
3 that cause the machine to :

4 retrieve the load data from a shared memory system; and

5 provide the load data to the private cache of the main core.

1 27. The article of claim 25, further comprising:

2 a plurality of machine accessible instructions, which if executed by a machine, cause the
3 machine to perform operations comprising:

4 providing load data for the load instruction from a shared memory system into the
5 private cache of the helper core.

1 28. The article of claim 25, further comprising:

2 a plurality of machine accessible instructions, which if executed by a machine, cause the
3 machine to perform operations comprising:

4 providing load data for the load instruction from a shared memory syste5 into the
5 private cache for each of a plurality of helper cores.

1 29. The article of claim 24, wherein:

2 the instructions that cause the machine to prefetch load data further comprise instructions
3 that cause the machine to :

4 retrieve the load data from a private cache of a helper core; and

5 provide the load data to the private cache of the main core.

1 30. The article of claim 24, wherein:

2 the instructions that cause the machine to prefetch load data further comprise instructions
3 that cause the machine to :

4 concurrently:

5 broadcast a request for the load data to each of a plurality of cores; and

6 request the load data from a shared memory system.

1 31. The article of claim 25, wherein:

2 the instructions that cause the machine to prefetch load data further comprise instructions
3 that cause the machine to:

4 provide, if the load data is available in a private cache of one of the plurality of cores,
5 the load data to the main core from the private cache of one of the plurality of cores ; and

6 provide, if the load data is not available in a private cache of one of the plurality of
7 cores, the load data to the main core from the shared memory system.

1 32. A system comprising:

2 a memory system that includes a dynamic random access memory;

3 a first processor, coupled to the memory system, to execute a first instruction stream;

4 a second processor, coupled to the memory system, to concurrently execute a second
5 instruction stream; and

6 helper threading logic to provide fill data prefetched by the second processor to the first
7 processor.

1 33. The system of claim 32, wherein:

2 the helper threading logic is further to push the fill data to the first processor before the
3 fill data is requested by an instruction of the first instruction stream.

1 34. The system of claim 32, wherein:

2 the helper threading logic is further to provide the fill data to the first processor from a
3 private cache of the second processor.

1 35. The system of claim 32, wherein:

2 the helper threading logic is further to provide the fill data to the first processor from the
3 memory system.

1 36. The system of claim 32, further comprising:

2 an interconnect that manages communication between the first and second processors.

1 37. The system of claim 32, wherein:

2 the memory system includes a cache that is shared by the first and second processors.